DOCUMENT RESUME

ED 255 560 TM 850 203

AUTHOR Gray, Peter J.
TITLE Microcomputers: Statistical Analysis Software. Evaluation Guide Number 5.
INSTITUTION Northwest Regional Educational Lab., Portland, OR. Research on Evaluation Program.
SPONS AGENCY National Inst. of Education (ED), Washington, DC.
PUB DATE [84]
CONTRACT 400-80-0105
NOTE 13p.
PUB TYPE Guides - Non-Classroom Use (055)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Software; *Evaluation Criteria; *Microcomputers; *Statistical Analysis; Summative Evaluation

ABSTRACT

This guide discusses six sets of features to examine when purchasing a microcomputer-based statistics program: hardware requirements; data management; data processing; statistical procedures; printing; and documentation. While the current statistical packages have several negative features, they are cost saving and convenient for small to moderate data sets when compared to mainframe computers. It is important to evaluate a statistical analysis program in terms of its versatility regarding the features needed most. (BS)

EVALUATION GUIDES

Guide Number 5

MICROCOMPUTERS: STATISTICAL ANALYSIS SOFTWARE

Peter J. Gray

The use of microcomputer and statistical analysis programs are discussed, including:

- Key Characteristics of Statistical Analysis Software
- Pros and Cons of Statistical Analysis Software
- Selecting the Right Software
- References

Research on Evaluation Program Nick L. Smith, Director

Northwest Regional Educational Laboratory
300 S.W. Sixth Avenue • Portland, Oregon 97204 • Telephone (503) 248-6800

KEY CHARACTERISTICS OF STATISTICAL ANALYSIS SOFTWARE

There are six sets of features that should be examined when considering the purchase of a microcomputer-based statistics program. These include hardware requirements, data management, data processing, statistical procedures, printing, and documentation. Each of these features is examined in more detail in the following discussion.

Data Management

Two elementary features of statistical programs are the kind and amount of data they can handle. In regard to the kind of data, the question is, can one enter integer, character (alphanumeric) and/or decimal data? Most programs do not permit all three kinds of input. And, regarding the amount of data a program can handle, it is important to ask, what is the total possible number of variables, cases, and data items (i.e., the product of variables and cases)? Limitations on the amount of data are due to a variety of factors. One is that a program can operate only on data located in the internal memory of a machine, that is, random access memory or RAM. Another factor is that a program can access data from only a single storage disk. Still another factor is the dimensions of the arrays that a program can handle.

Programs vary considerably in their data management capabilities. Therefore, in order to decide among programs, it is necessary to determine one's own requirements for type and amount of data. An article by Carpenter, Deloria, and Morganstein (1984) describes these and other characteristics of 24 different statistical programs. Much of the detailed information in this column regarding the capabilities of programs was obtained from their article. A word of caution, as with any article, there may be a problem of information being out of date, especially in as fast-changing an area as statistical software. Readers are advised to contact individual producers or manufacturers to get the latest information on packages. Some programs, such as Statpro, also offer technical references, reviews, and names of people who are currently using the product. These are good sources of information, although they may be biased toward a particular product.

Actual data management starts with the entry of data into the program. The way that a program stores files internally influences both data entry and data transfer. If a program stores files in a unique way, it may be difficult to use data other than those entered directly in the program via the keyboard, and it will be difficult to share those data with another program (e.g., graphics, word processing). This is

especially true if the storage format is not well documented since this inhibits one's ability to change the data format to make it compatible with other file structures. The issues surrounding documentation will be discussed later in this column.

There are several ways that data can be entered into a program. One is directly from the keyboard using the program itself. Data can also be transferred into a program from other programs via the Data Interchange Format (DIF) used by VisiCalc, for example, by using the standard ASCII files created from keyboard entries by other programs, or from specially formulated files created by special editors or by mainframe programs. Deciding among programs in regard to their mode of data entry and file structure will depend largely on whether files are likely to be transferred among programs. If files are typically shared, then compatibility is a must. If this is not the case, then ease of data entry is the main criteria for comparing programs.

Two data entry procedures exist, case-by-case entry, where all variables for each case are entered at the same time, and variable-by-variable entry, where all cases for each variable are entered as a group. Better programs give you a choice between these two procedures. However, a single data structure should be used throughout the program. That is, the cases should be the rows and the variables the columns, or vice versa, consistently throughout a program.

How a program handles errors in data entry or the entry of commands is an important factor in its ease of use. Seventeen of the 24 programs reviewed by Carpenter, Deloria, and Morganstein (1984) were rated fair or poor in the extent to which they allowed recovery from errors, explained an error, and/or indicated the source of error. One-third had poor error handling capabilities. Such programs will typically just stop operating, and the only way to proceed is to start the program all over again, which may mean re-entering all of the current data.

Similarly, in programs with poor error handling, it may be difficult to get out of a data management, data processing, or statistical procedure that was erroneously chosen. In the best case, one can simply "back out" of the procedure by pressing the ESCape key, for example. In the worst case, the program will leave you no choice but to exit entirely to recover from the error.

Once data are in the program, procedures for maintaining them are needed. At a basic level, maintenance includes adding new data, correcting erroneous data, and deleting unneeded data. An editor built into a program can be especially convenient for

these purposes. This makes it unnecessary to exit the program to enter data, or to make changes in the data, and then re-enter the program to continue with data processing and statisical analysis.

Data Processing

There are a number of related program features to look for regarding the processing of data and their statistical manipulation. Data processing covers the selection of cases, the joining of files, the sorting of files, and the transformation of data. Often the analysis of one or more subsets of the total number of cases is desired. There are several ways to select cases for these purposes. The most powerful way is to use Boolean AND and OR conditions between ranges of variables. For example, you may wish to select two discrete groups of students for comparison, such as students with math scores from 1 to 3 AND reading scores from 1 to 3, OR students with math scores from 8 to 10 AND reading scores 8 to 10. Other options include selecting just a single range of a variable or a set of specific values. The procedures currently used are the best guide in determining how sophisticated a program's ability to select cases needs to be.

Reviewing current procedures will also help determine which of the other following data processing features are desired. For example, the ability to join new cases, or new variables for current cases, with an already existing file may be important if you have a continuously changing data base. Sorting a file is sometimes necessary to perform subsequent statistical analyses; therefore, depending on the type of analyses that you are interested in, this feature may be more or less desirable.

The transformation of data is also frequently required for statistical analyses. Transformations include single variable operations such as logical operations, adding a constant, and taking logs, as well as mathematical computations involving more than one variable. Transformations may be used to create new variables and to generate random variables for use in simulations. In addition, the weighting of variables is often achieved through a transformation process if it can not be accomplished directly during statistical analysis. Past experience is the best source of information for determining the desirablity of these features.

Statistical Procedures

Programs vary considerably in the types of statistical procedures they include and in the variety of statistics regarding any one type. Not all programs contain all types of

procedures, and programs include different specific procedures within a type. Figure 1, adapted from the tables in Carpenter, Deloria, and Morganstein (1984), shows a sample of the types and variety of statistical procedures that might be found in a package. According to these authors, the procedures most frequently included are summary statistics (especially arithmetic means, variance and standard deviation), simple and multiple regression analyses, paired and groups t-tests, and various N-way ANOVAs. Very few programs appear to have nonparametric statistics other than the Spearman Rank-Order Correlation. In addition, of the 24 programs they reviewed, few offer time series procedures. Therefore, careful attention to the procedures included in a given package is advised!

Figure 1
Sample of Statistical Procedures*

Descriptive Statistics	
Frequency	Mean
Median	Mode
Percentile	Range
Minimum/Maximum	Variance
Standard Deviation	Skewness
Ranks	Kurtosis
Nonparametric Statistics	
Friedman Two-Way ANOVA	Kendall's Tau
Mann-Whitney U	Kolomogorov-Smirnov
Kruskal-Wallis H	Spearman Rank-Order Correlation
Wilcoxon Signed Rank	Wald-Wolfowitz Runs
Chi-Square Tests (e.g., goodness of fit, log-linear models)	Kendall's Coefficient of Concordance
	Contingency tables/cross-tabs
Linear Models	
Regression; simple, multiple, polynomial, stepwise	T-tests; paired, groups
General Linear Model	ANOVA; N-way, contrasts, unequal cell size, random effects
Factor Analysis	Discriminant Analysis
Time Series	
ARIMA	Two-Stage Ordinary Least Squares
Cochrane-Orcutt	Serial Correlation
Moving Averages	(Autocorrelation Coefficients)

* Adapted from Carpenter, Deloria, and Morganstein (1984)

Statistical accuracy and the time it takes to compute a given solution are issues apart from which statistics are available. In its simplest form, statistical accuracy is whether a program can calculate the correct answer. The precision of the statistic can be a problem on microcomputers because of rounding error due

to their limited ability to handle numbers with many digits. In addition, inaccuracy may result from a problem with the algorithm used by the program to calculate the statistic or a problem regarding the condition of the data matrix.

The time it takes to compute a statistic will depend on how the data are handled. In some programs all of the data are placed in the computer's internal memory (RAM). This greatly enhances the rate of processing, since the only limit is the internal processing speed of the computer. However, it limits the amount of data to that which can fit in RAM.

By increasing the amount of internal memory, more data can be accommodated. External memory on either flexible or hard disks can also increase the amount of data that can be accessed, but it will limit the speed in comparison to internal memory-based programs, since the mechanical process of going to the disk to read data, transferring it into RAM, and then processing it all takes time.

Internal processing speed varies considerably from machine to machine. Certain machines' speed in processing numbers can be enhanced by using one of the many packages based on the Intel 8087 math co-processor chip. With this chip and related software, speed of processing can be increased up to 180 times and accuracy increased to 18-digit precision.

With some programs, machines, and data sets, it is necessary to allow several hours or even an overnight period for processing to occur. However, this may not be any longer than is necessary to complete a similar task on a heavily used mainframe time-share system. And, in a short time, a number of such runs may pay for the microcomputer in terms of cost and convenience. The best advice is to have a data set typical of your research that can be used as a benchmark for judging a program's accuracy and speed.

Printing

There are times throughout the process of using a statistical program that printing is desired. When data are being entered, for example, one may wish to get a printed case-by-case or variable-by-variable summary in order to verify their accuracy. Printed results of computations can be helpful, especially those of intermediate processes such as a regression equation, which may be used in other computations. And, of course, the results of analyses should be printed in a format that is consistent with conventions (e.g., contingency table, ANOVA table) and should be clearly labeled for easy interpretation.

Beyond the printing of individual results, it is often useful to present a graphic picture in the form of a scatter plot, histogram, and so on. These may then be combined with the results of analysis to form a report. The very best programs allow control over (1) the placement of titles, labels, and other textual information (e.g., headings and footnotes), (2) the selection among various types of graphic depictions of results (bar, line, pie, scatter-plot, histogram, 2-D and 3-D) and control over their scaling, and (3) the storage of completed reports for later use. An alternative to having these capabilities built into a program is being able to easily transfer data to another program that has the specific reporting capabilities desired.

Written Information

Unfortunately, information about data management, data processing, statistical procedures, speed, accuracy, and so on, are unevenly treated in the written information accompanying most programs. This is evident in the fact that the number of pages in the documentation of the programs reviewed by Carpenter, Deloria, and Morganstein (1984) ranged from 12 to 381. This unevenness is true for both the written information about how to use most programs and about their technical aspects. The documentation for a program can be a major factor in facilitating or hampering its use. Good documentation for statistical software will at least provide some description of (1) the way the software is organized, (2) some basic information about each feature of the program, and (3) more detailed information about the statistical procedures and when different procedures might be selected.

Welcome additions include several features that can make a program easy to use. These might include a set of written examples (e.g., sample runs). Both tutorial and "help" features regarding how to use each part of the program can also facilitate the use of a program. A tutorial is typically a program in itself designed to teach the user about a feature by working through it step by step at the computer. If a mistake is made, feedback is provided that can help a person understand what the mistake was and why it caused a problem. A "help" function is not a separate program, but instead is build right into a statistical program. It is used when a question arises regarding a particular feature. At that point, additional information is given on the screen to help one better understand what the feature is and how to use it, but no tutorial process is involved, just information.

Clear illustrations of what will be presented on the screen at each step in using a program can be a great help both in initially learning a program and in getting the most out of those

features that are used only occasionally. Illustrations or examples of the kinds of results one can expect to appear on the screen at each step of the program can be very reassuring. Similarly, examples showing how to create various printed versions of results, and what they will look like, can improve the ease with which this often-complex feature is mastered.

Exemplary documentation will include the particular algorithm used in each analysis so that the user can be fully aware of its assumptions. The best programs will also document instructions for modifying formulas to better meet particular situations. Tables of contents, indexes, and an alphabetized summary of commands are all features that should be expected in a package, but which are sometimes not included.

PROS AND CONS OF STATISTICAL ANALYSIS SOFTWARE

Most statistical packages have a long way to go before they reach the professional appearance and ease of use of word processing, data base management, spreadsheet, and other generic business software. Some of the most negative features of the packages reviewed by Carpenter, Deloria, and Morganstein (1984) were:

- cumbersome nature of handling large data sets and in some cases the inability to handle such sets,
- potential need to buy more than one package to get all the statistical procedures desired,
- poor documentation regarding the number of cases and variables allowed in relation to the RAM available,
- typically slow processing speed for larger data sets,
- the need to enter the same information about number of cases, number of variables, file names, etc., again and again when going through related operations,
- the need to go through all steps in a menu when doing the same procedure with different variables,
- changing the meaning of key commands without alerting the user,
- the lack of escape options in case a mistake is made either in entering data or selecting a menu item,

- the tendency of programs to crash when an error was made, with only uninterpretable messages about the cause.

It is important to remember that most microcomputer-based statistical packages have only been on the market a relatively short time. Consumer pressure and competition will undoubtedly help to increase the quality of programs over the next several years.

On the positive side, the cost savings and convenience of using the current crop of statistical packages for small to moderate data sets make them an attractive alternative to mainframe computer use. To the extent that both input and output from a program can be shared with other programs, such as generic business packages or mainframe programs, microcomputer-based statistical programs offer a substantial enhancement of one's computing repertoire. In addition, there are individual programs that bring the power and variety of mainframe computer programs to the micro, a very attractive possibility. And, as is apparent from our software application section, such power and variety can offset many of the limitations of poor documentation, difficult use, and slow processing speed.

The only way to decide which package is the right one for you is to think about the features described in this article in relation to the research you do, to talk with others who have experience doing similar research using particular packages, and then to try out a variety of packages to see for yourself which features are most important.

SELECTING THE RIGHT SOFTWARE

In one sense, hardware requirements are the first characteristic of a program that should be considered and, in another sense, they are the last. From a realistic point of view, the first criteria for selecting a program is whether it will run on a machine you already have, or on a machine that you feel you can afford to buy for statistical and other purposes. However, within these general constraints, hardware becomes a secondary consideration, because there is a variety of good programs to choose from for most of the popular and widely used machines with operating systems such as Apple DOS 3.3, IBM-PC DOS or MS-DOS, and CP/M. Within each group, programs vary in terms of their sophistication and cost and in terms of the specific hardware system characteristics that they require.

In summary, if a commitment has already been made to purchase a particular machine, or if there are special budget limitations, hardware-related requirements are the first features of a program that should be considered. However, if there are no rigid constraints, it is best to ignore these requirements for the time being and move on to the other more substantive features of statistical analysis programs.

It is important to evaluate a program in terms of its versatility regarding those features you need most. Selection may come down to the program(s) with the best ratings on those features of greatest importance as opposed to those with he best over-all ratings. This notion of the highest ratings on the most important features is worth considering. Sometimes pricing, especially in regard to multiple copies, is the deciding factor among programs of generally equal ratings. In other cases it may be that speed, error handling, and versatility (i.e., program performance) is more important than either ease of use or support. Therefore, lower ratings in these areas would not disqualify a program if it was a strong performer.

Using the information provided in this guide will help you to judge the quality of individual programs. The procedures also provide a way to compare programs in a consistent manner.

Any combination of features is possible. Selection should be based, therefore, on a consideration of the combination of features most desired for the types of tasks to be performed using the program.

In order to make a sound choice:

1. Describe your use(s) - what will you use the program for?

2. Identify the features you need - what do you want to be able to do?

3. Plan ahead for new needs - what are you likely to want a year from now?

4. Consider constraints - What price range, hardware (e.g., machine type, printer features) and user preferences are you limited by?

5. Put features into a rough priority list - which are the most, somewhat, and least important features?

6. Try out and compare products - which ones have the features you need and want within your constraints?

7. Remember support - will there be someone you can talk to if there are problems after you buy the program?

REFERENCES

Callamaras, P. (1983, October). Statistical software: A buyer's guide to seven major types of statistic packages. Popular Computing, 206; 208; 210; 211-214; 216; 218-220.

Carpenter, J., Deloria, D., & Morganstein, D. (1984, April). Statistical software for microcomputers: A comparative analysis of 24 packages. BYTE, pp. 234-236, 238-252, 254, 256, 258, 260, 262, 264.

Hsu, T., & Nitko, A. (1983, June). Microcomputer testing software teachers can use. A paper presented at the 1983 Education Commission of the States Large-scale Assessment Conference, Boulder, CO.

Lachenbruch, P. A. (1983, November). Statistical programs for microcomputers. Byte, 8(11), 560-563, 566-568, 570.

Neffendorf, H. (1983, February). Statistical packages for microcomputers: A listing. The American Statistician, 37(1), 83-86.

RECENT GUIDEBOOKS IN THIS SERIES

No.	Title
1	Microcomputers and Evaluation
2	Cost-Outcome Analysis: Measuring Costs
3	Microcomputers: Word Processing
4	Cost-Outcome Analysis: Measuring Outcomes
5	Microcomputers: Statistical Analysis Software
6	Investigative Journalism Techniques
7	Microcomputers: Data Base Management Software
8	Committee Hearings: Their Use in Evaluation
9	Microcomputers: Spreadsheet Software
10	Methods of Product Evaluation

Printed by the Northwest Regional Educational Laboratory, a private nonprofit corporation. The work upon which this publication is based was performed pursuant to Contract No. 400-80-0105 of the National Institute of Education. It does not, however, necessarily reflect the views of that agency.

Research on Evaluation Program
Northwest Regional Educational Laboratory
300 S. W. Sixth Avenue
Portland, OR 97204